Statistical and Graph Theoretical Approaches to Semantic Tagging of Unstructured Text for the BKC

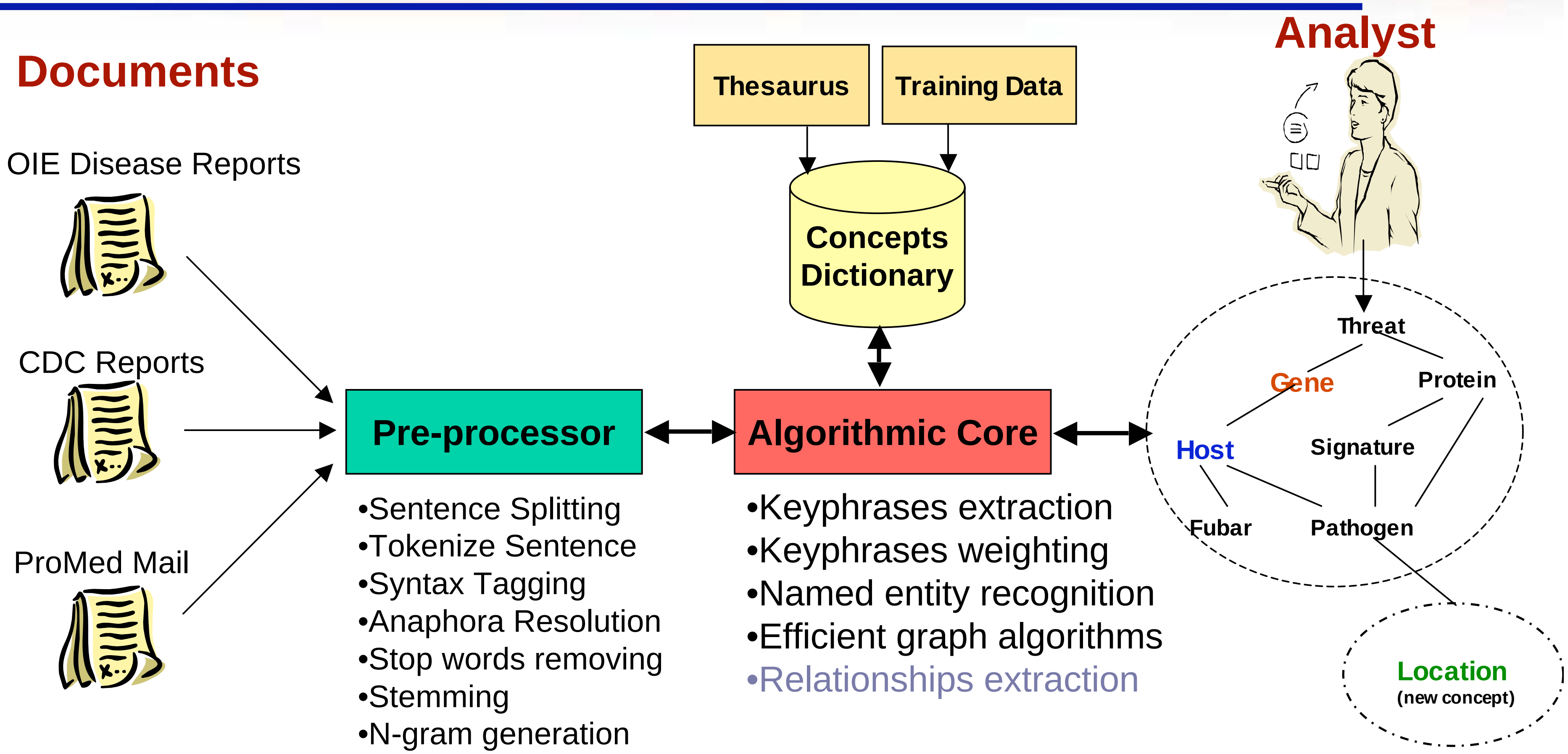
DHS Advanced Scientific Computing Program

Nagiza F. Samatova
(samatovan@ornl.gov)
Oak Ridge National Laboratory

Motivation and Goals

- Over 100 data sources have been initially identified for inclusion in the BKC; most of them contain rich information in the form of free text
- The amount of relevant information is increasing daily making manual reading and curation infeasible
- **Our goals:**
  - Provide methods for automatic extraction and semantic tagging of important information from free text to make it accessible through the BKC semantic graph
  - Facilitate efficient querying over the semantic graph

System Overview



Analyst

Documents

OIE Disease Reports

CDC Reports

ProMed Mail

Thesaurus

Training Data

Concepts Dictionary

Pre-processor

Algorithmic Core

- Sentence Splitting
- Tokenize Sentence
- Syntax Tagging
- Anaphora Resolution
- Stop words removing
- Stemming
- N-gram generation

- Keyphrases extraction
- Keyphrases weighting
- Named entity recognition
- Efficient graph algorithms
- Relationships extraction

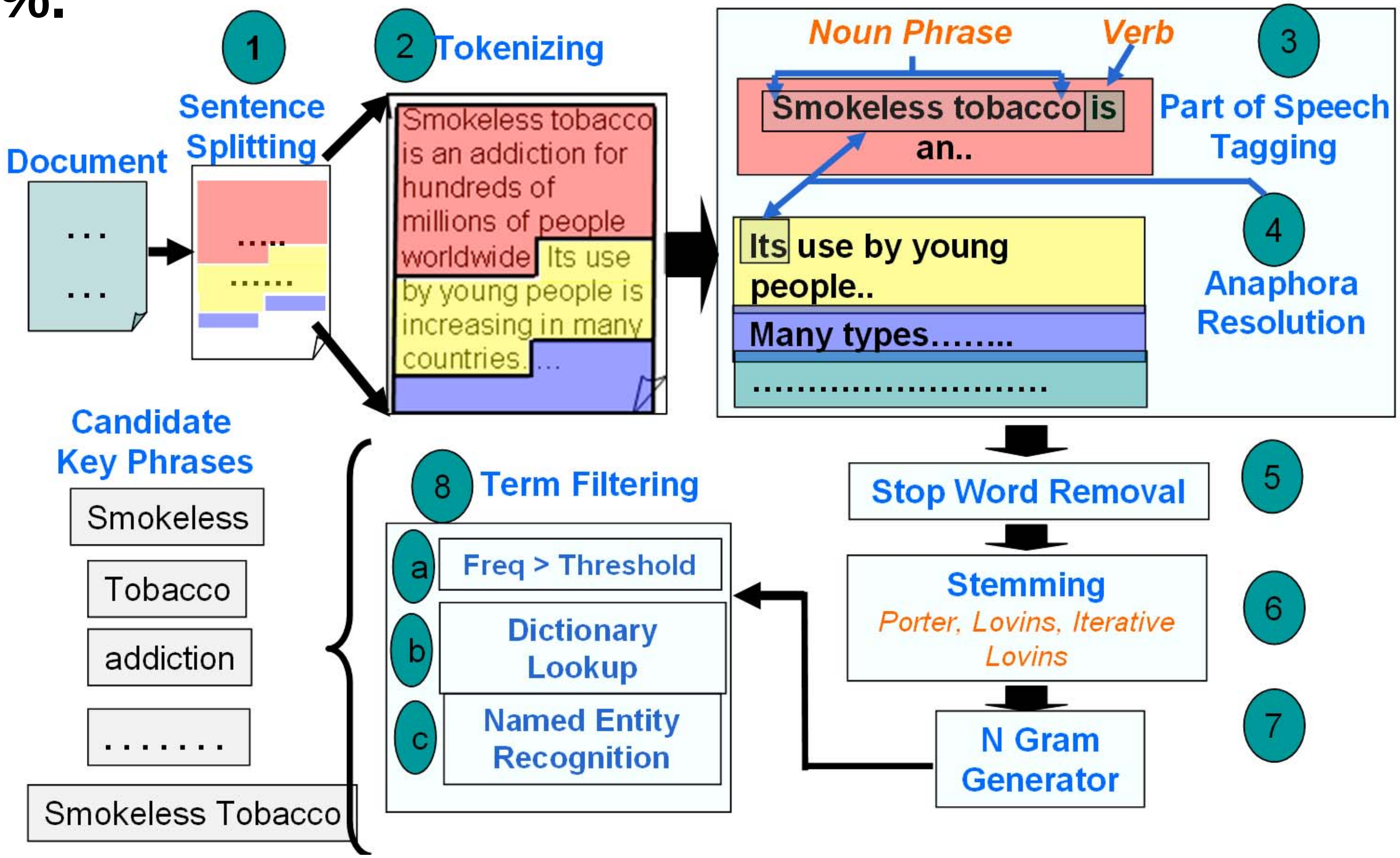
Foot and Mouth Disease

A virus of the family **Picornaviridae, genus *Aphthovirus***. Seven immunologically distinct serotypes: A, O, C, SAT1, SAT2, SAT3, Asia1.

Hosts: **Bovidae (cattle, zebus, domestic buffaloes, yaks), sheep, goats**, swine, all wild ruminants and suidae. **Camelidae (camels, dromedaries, llamas, vicunas**) have low susceptibility. FMD is endemic in parts of **Asia, Africa, the Middle East and South America** (sporadic outbreaks in free areas)

Intelligent Text Preprocessing within BKC

Text preprocessing is critical since it can improve the performance of text analysis algorithms by 15-20%.



1 Sentence Splitting

Document

… …

……

……

2 Tokenizing

Smokeless tobacco is an addiction for hundreds of millions of people worldwide Its use by young people is increasing in many countries…

Noun Phrase

Verb

3 Part of Speech Tagging

Smokeless tobacco is an..

Its use by young people..

Many types……..

……………………….

4 Anaphora Resolution

5 Stop Word Removal

6 Stemming

Porter, Lovins, Iterative Lovins

7 N Gram Generator

8 Term Filtering

a Freq > Threshold

b Dictionary Lookup

c Named Entity Recognition

Candidate Key Phrases

Smokeless

Tobacco

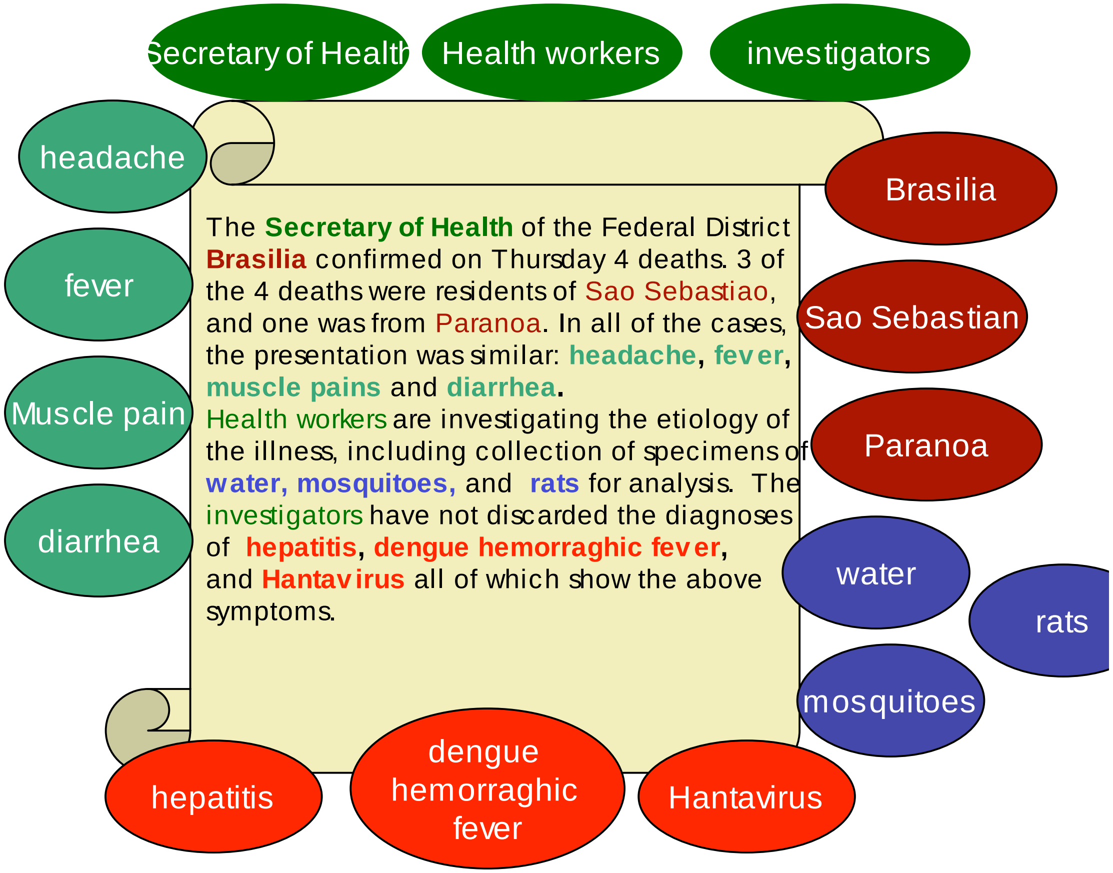
addiction

……..

Smokeless Tobacco

Keyphrases Extraction and Weighting

- Keyphrases extraction is often the first step towards extracting information from free text documents.
- Keyphrases provide a reasonable understanding of the document content.
- Appropriate weights give the relevance of a document to a particular topic.



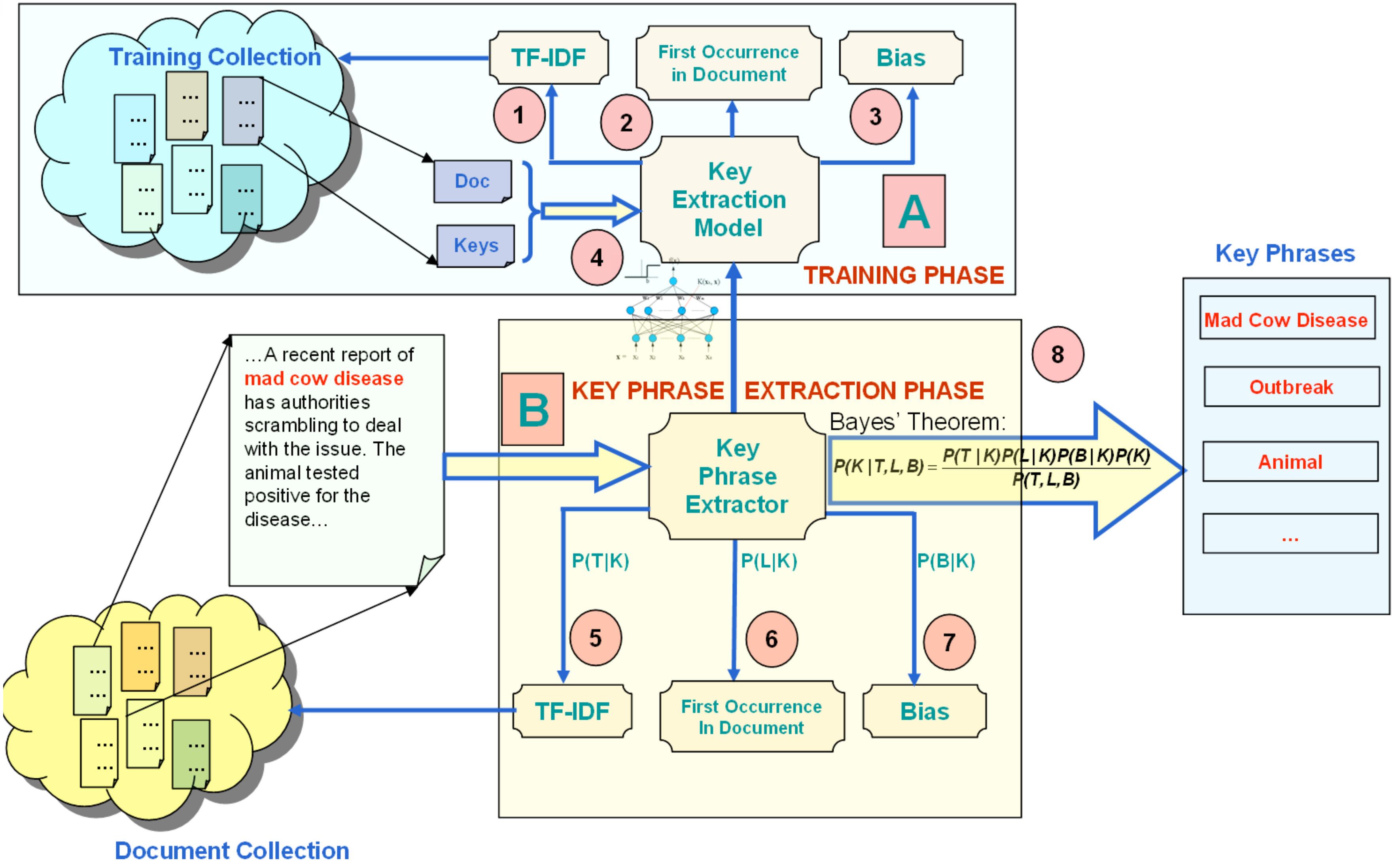
The **Secretary of Health** of the Federal District **Brasilia** confirmed on Thursday 4 deaths. 3 of the 4 deaths were residents of Sao Sebastiao, and one was from Paranoa. In all of the cases, the presentation was similar: **headache**, **fever**, **muscle pains** and **diarrhea**. Health workers are investigating the etiology of the illness, including collection of specimens of **water, mosquitoes,** and **rats** for analysis. The investigators have not discarded the diagnoses of **hepatitis**, **dengue hemorraghic fever**, and **Hantavirus** all of which show the above symptoms.

Approaches to Keyphrases Extraction –
Corpus-Dependent and Corpus-Independent Methods

- A **corpus dependent** approach can be very useful when documents come from the ***same source*** and usually pertain to ***related topics***.
  - We developed a Naïve Bayesian classifier method for situations that allow a corpus-dependent approach.
  - Utilizes **domain-specific knowledge** relevant to BKC as a basis for the bias in the Corpus Dependent Method.
  - Provides marked improvement in the observed keyphrase extraction.
  - Allows identification of documents relevant to BKC without forcing inclusion of documents simply because they contain a related term.

- A **corpus independent** approach can be very useful if the source of the documents is not very consistent and the documents could belong to ***a variety of domains***.
  - We developed a term co-occurrence based algorithm for situations that call for a single-document method.

Corpus-Dependent Keyphrase Extraction



…A recent report of mad cow disease has authorities scrambling to deal with the issue. The animal tested positive for the disease…

$$P(K \mid T,L,B) = \frac{P(T \mid K)P(L \mid K)P(B \mid K)P(K)}{P(T,L,B)}$$

Corpus-Independent Keyphrase Extraction

Less frequent but important words undetected by TF method

Smokeless tobacco is an addiction for hundreds of millions of people w orldw ide. **Use** by young people is increasing in many countries. Many types of **smokeless tobacco** are marketed for oral or nasal **use**. All contain nicotine and nitrosamines. DNA and haemoglobin adducts are commonly detected in **tobacco users**

Tobacco users are exposed to differing levels of nitrosamines. These are formed mainly by nitrosation of nicotine and other **tobacco** alkaloids during the curing and processing of **tobacco**, and additional amounts are formed during smoking……

1 Top 30% words filtered by TF method

2 Term Clustering

3 Co-occurrence Distribution Significance Score (X^2)

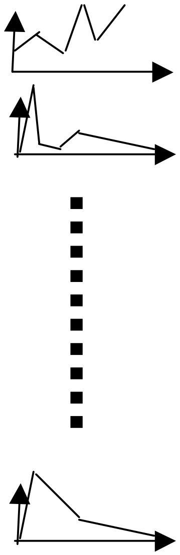
4 N-Gram collapsing

$$\chi'^2(w) = \sum_{c\in G}\left\{\frac{(freq(w,c)-n_w p_c)^2}{n_w p_c}\right\} - \max_{c\in G}\left\{(\frac{(freq(w,c)-n_w p_c)^2}{n_w p_c}\right\}$$

Frequent terms

All terms	tobacco	use	addict	…	X^2
tobacco	-	6	3	11	132
use	6	-		7	30
nicotine	8	5	5	2	342
expose	5	7	1	4	23
…	…	…	…	…	…
direct expose	2	5	1	7	258
smokeless tobacco	9	4	2	0	545

Co-occurrence matrix



Terms Clustering – Similarity Measures

Distribution-based Similarity

- Two terms are considered to be similar if they have similar co-occurrence distribution of co-occurrence with all the other terms.

- **Jensen-Shannon divergence value** of two terms indicates the distribution similarity.

$$J(w_1, w_2) = \log_2 2 + 1/\,2 \sum_{w`\text{€}G} \{h(P(w`|w_1) + P(w`|w_2)) - h(P(w`|w_1)) - h(P(w`|w_2))\}$$

Where

$h(x) = -x \log x$, $P(w`|w1) = freq(w`,w1) / freq(w1)$

Pair-wise Similarity

- Two terms are assumed similar if they co-occur frequently.
- Pair-wise similarity is measured by **mutual information**

$$M(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)\, P(w_2)}$$

Terms Clustering

Averaging hierarchical model based clustering scores

1

Column terms

- tobacco
- use
- …
- addict

2

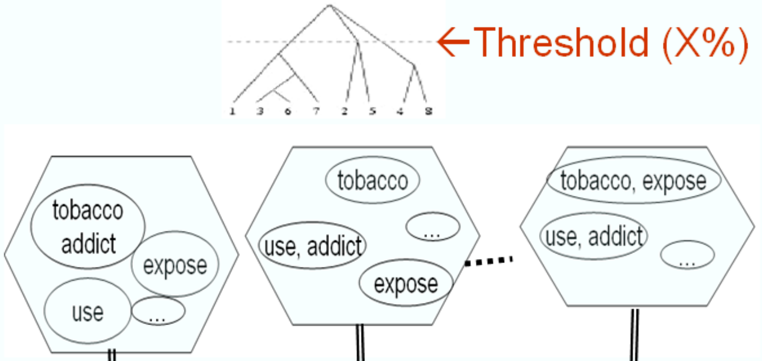
Similarity Measure Matrix

(Jensen Shannon divergence OR Mutual information)

	tobacco	use	…	addict
tobacco	-	0.1	0.3	0.2
use	0.1	-	0.2	0.8
…	0.3	0.2	-	0
addict	0.3	0.8	0	-

3

Hierarchical Clustering



The choice of clustering Threshold dramatically affects the results.

Averaging over 10%-30% in 5% increments provides robust scores.

Co-occurrence distribution significance

4

Row terms

Clustered Column terms

	tobacco	use	addict	…
tobacco	-	6	3	11
use	6	-		7
nictoine	8	5	5	2
expose	5	7	1	4
…	…	…	…	…
direct expose	2	5	1	7
smokeless	9	4	2	0

3

Co-occurrence Distribution significance

	tobacco	use	addict	…
tobacco	-	6	3	11
use	6	-		7
nictoine	8	5	5	2
expose	5	7	1	4
…	…	…	…	…
direct expose	2	5	1	7
smokeless	9	4	2	0

3

Co-occurrence Distribution significance

$\sum$

5

Averaging scores over different clusters

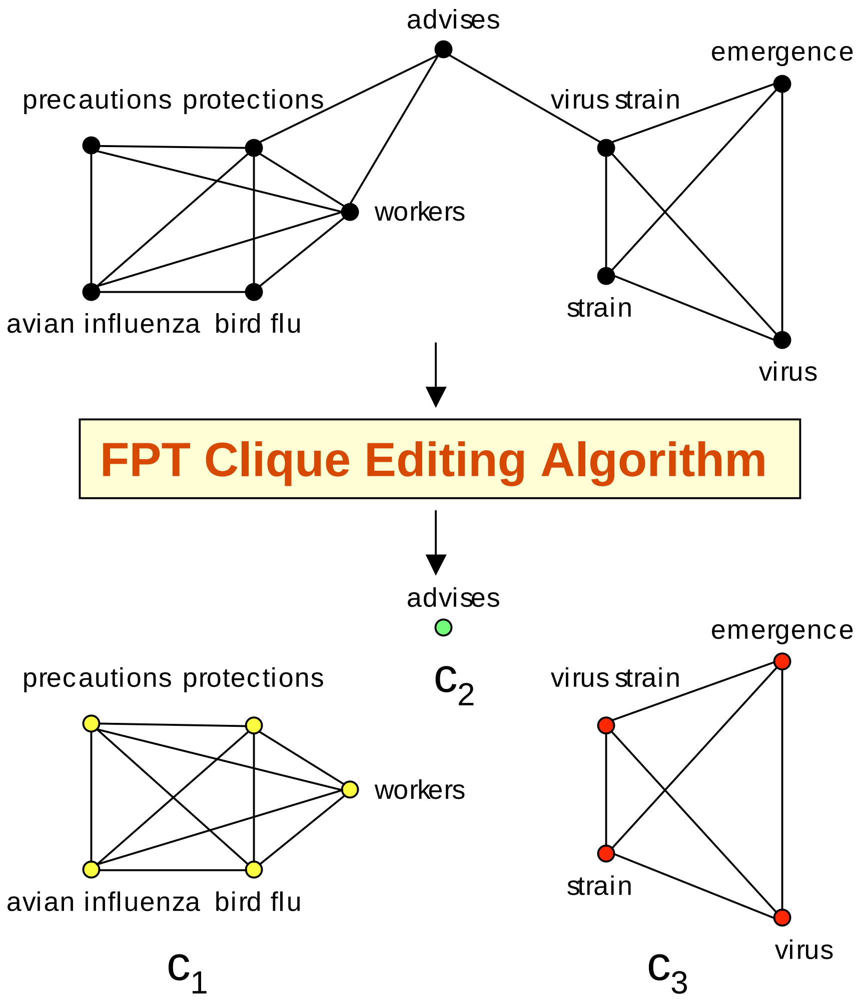
6

Key words

- Smokeless tobacco
- nicotine
- expose

Clique-based Term Clustering

- The choice of clustering Threshold dramatically affects the results. Averaging partially solves this problem.
- Still, hierarchical clustering assigns each term to a single cluster – no overlaps. However, latent semantic meaning of terms should allow terms belong to multiple clusters.
- We developed a form of clique-based clustering based on our efficient FPT clique editing algorithm.
- <u>Benefits</u>:
  - No need to *a priori* specify the number of clusters (reducing the error due to Thresholding)
  - Overall quality of clusters is better or comparable with the averaging method
  - Comparable computational time on small/medium documents with the averaging method

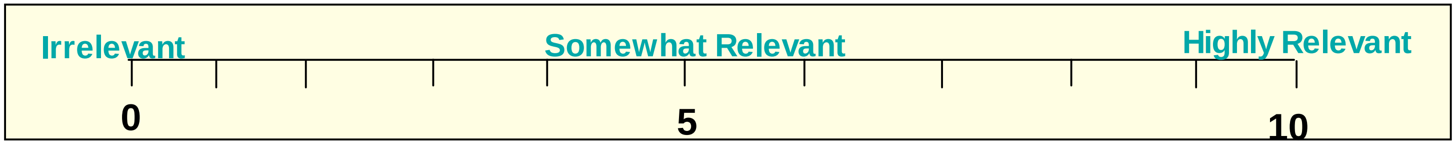


Evaluation of Keyphrases Extraction Methods

Document Collection:

Document Set	No of Documents
Aliweb	6
CSTR	12
Journal	6

Evaluation Method:



- **Top 15** keyphrases extracted by each algorithm were selected for evaluation

- **Individual Keyphrase quality** – Each keyphrase was scored according to its relevance to the document

- **Topic Coverage** **–** Entire keyphrase set was evaluated for coverage of topic(s) in the document

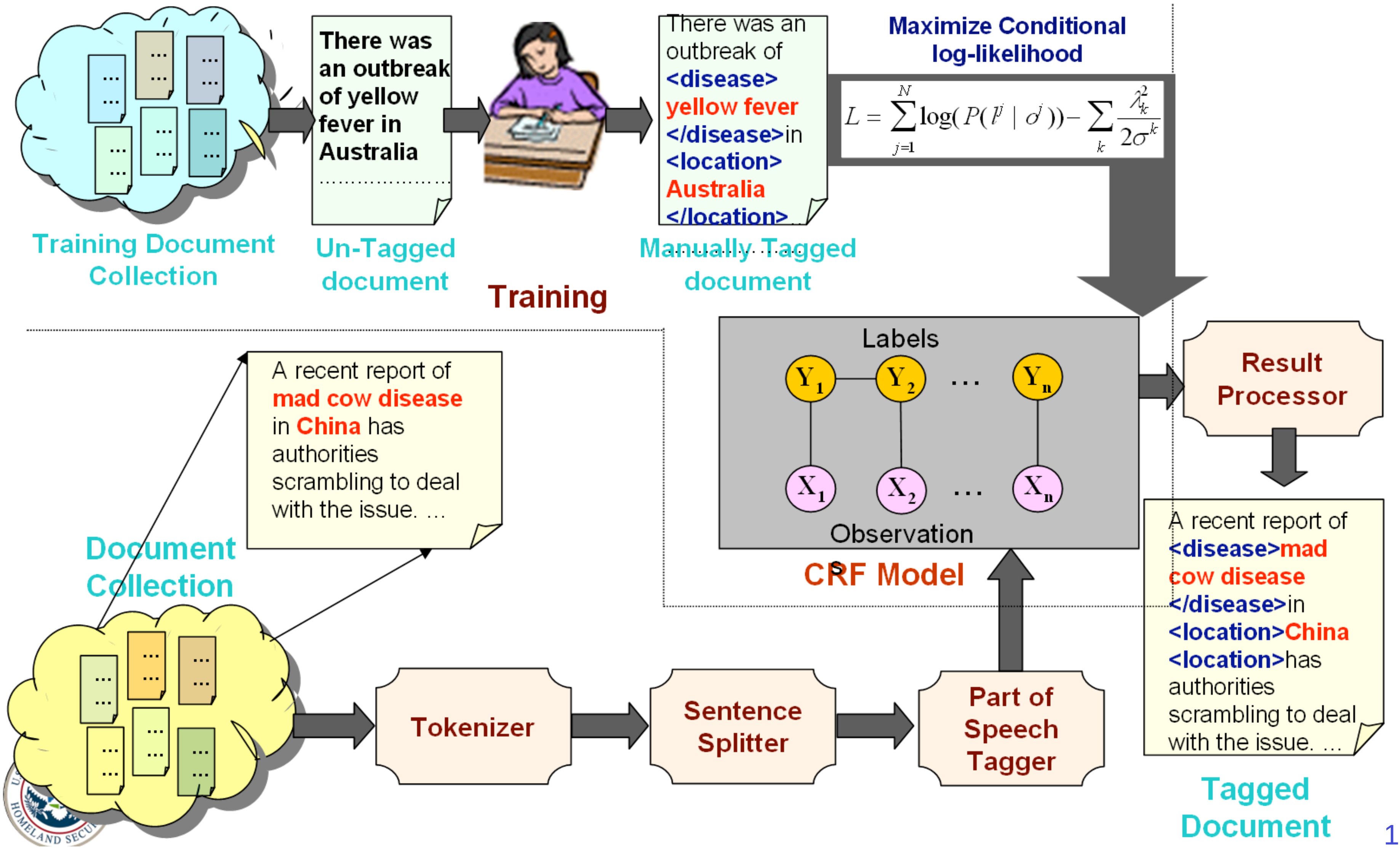
Results – *Manual* Evaluation of Key Phrases
Based on independent evaluation by 6 users

Algorithm	Keyphrase Quality			Topic Coverage		
	Average	Std Dev.	Avg. Rank	Average	Std Dev.	Avg. Rank
Author Assigned	5.8	1.7	9	5.9	1.2	6.4
Corpus Dependent (with Domain Bayes)	4.9	1.2	8	6.6	0.6	8.4
Corpus Dependent (no Domain Bayes)	4.7	1.3	6.8	6.4	0.7	7.4
TF-IDF	4.6	1.3	5.9	5.9	1.2	6.4
TF	4.1	1.5	4.4	5.2	1.1	4.2
Corpus Independent	4.5	1.4	5.8	5.8	1.3	6.4

- Corpus Independent algorithm compares very well with Corpus Dependent ones. The results are very much identical to TF-IDF method.
- Corpus Independent algorithm could extract more human readable phrases than TF or TF-IDF method.
- Corpus Independent method outperforms TF method that is also a corpus independent method in all respects.

Named Entity Recognition Pipeline within BKC

Names, Dates, Locations, Diseases, Bacteria, Proteins, …



Maximize Conditional log-likelihood

$$L = \sum_{j=1}^{N} \log(P(y^j \mid x^j)) - \sum_{k} \frac{\lambda_k^2}{2\sigma^k}$$

There was an outbreak of yellow fever in Australia ……………………

There was an outbreak of <disease> yellow fever </disease>in <location> Australia </location>……………

A recent report of mad cow disease in China has authorities scrambling to deal with the issue. …

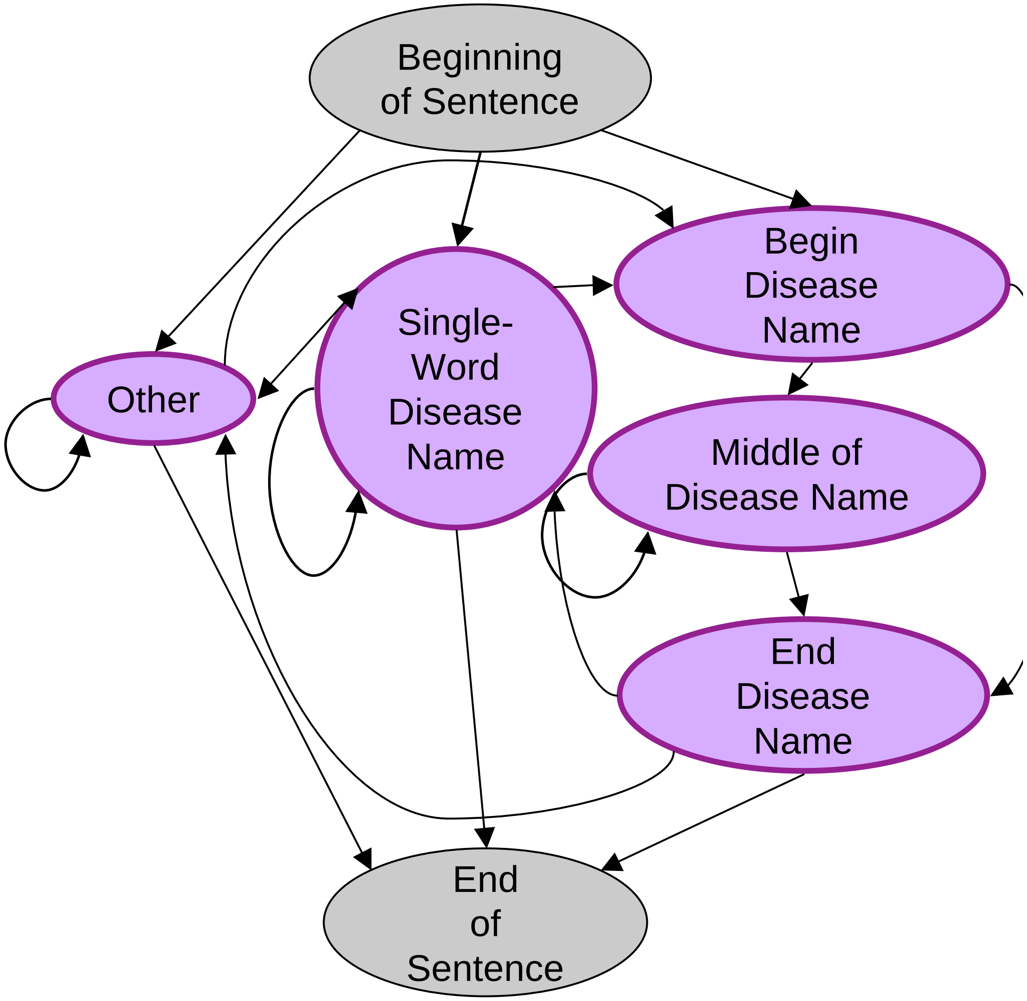
A recent report of <disease>mad cow disease </disease>in <location>China <location>has authorities scrambling to deal with the issue. …

Disease Tagging

- A **Conditional Random Field** based model allows us to utilize expert knowledge without worrying about overlapping features.

- Combines knowledge such as the following in our feature set:
  - known disease names
  - common words that end disease names
  - common orthographic endings of disease names
  - Latin and Greek roots in words
  - parts of speech

Intuitive Representation of Disease Mode



Performance Evaluation Results

Entity	Precision	Recall	F Measure
Disease*	77.73	73.04	75.31
Species¶	94.35	92.59	93.46
Genus¶	92.06	87.63	89.79

* Training performed using 250 ProMED mail documents;

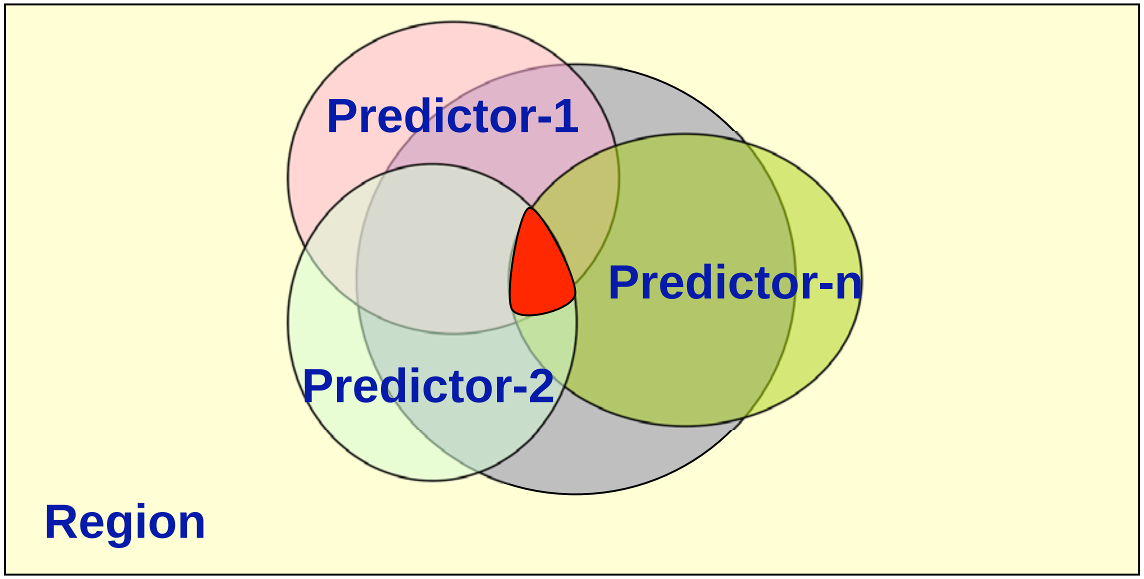
Testing performed on 100 separate ProMED mail documents.

¶ Number of training Documents: 100 ProMED email documents

Number of testing Documents: 47 ProMED email documents

Named Entity Recognition using Meta-Learning Techniques

To make use of the *existing* tools for Named Entity Recognition by exploiting non-overlapping regions of predictions to *improve performance* for predicting Protein names.



🟨 **Region under consideration**

⬛ **Protein Names under consideration**

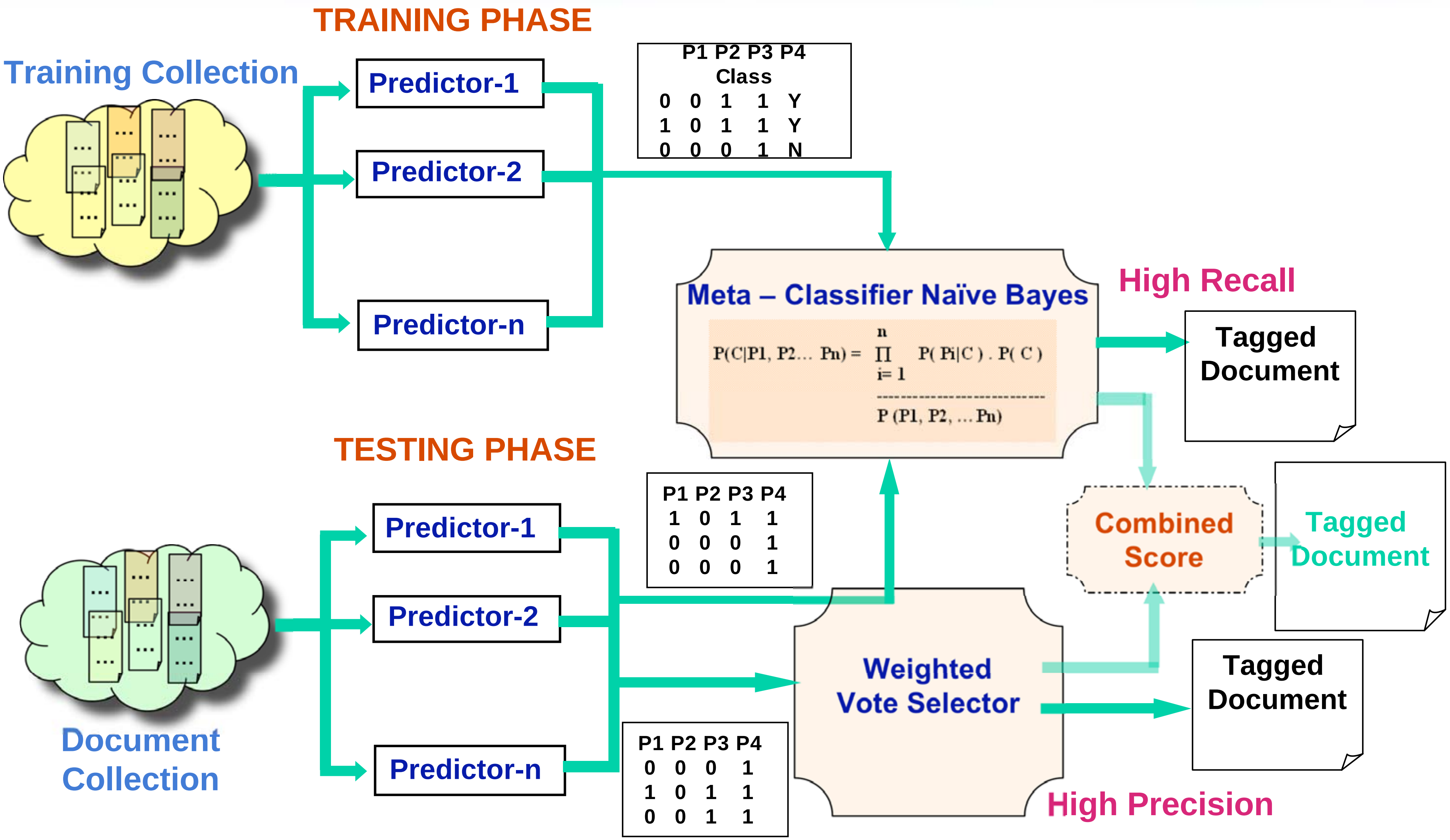
🟪 **Predictions by Predictor-1**

🟦 **Predictions by Predictor-2**

🟩 **Predictions by Predictor-n**

🟥 **Overlapping Predictions**

Meta-Learning and Weighted Voting Based Protein Named Entity Recognition



P1	P2	P3	P4	Class
0	0	1	1	Y
1	0	1	1	Y
0	0	0	1	N

$$P(C|P1, P2\ldots\ Pn) = \frac{\prod_{i=1}^{n} P(\,Pi|C\,)\,.\,P(\,C\,)}{P\,(P1,\,P2,\,\ldots\,Pn)}$$

P1	P2	P3	P4
1	0	1	1
0	0	0	1
0	0	0	1

P1	P2	P3	P4
0	0	0	1
1	0	1	1
0	0	1	1

5-fold Cross-Validation Results

Pasta Data (61 Medline abstracts)

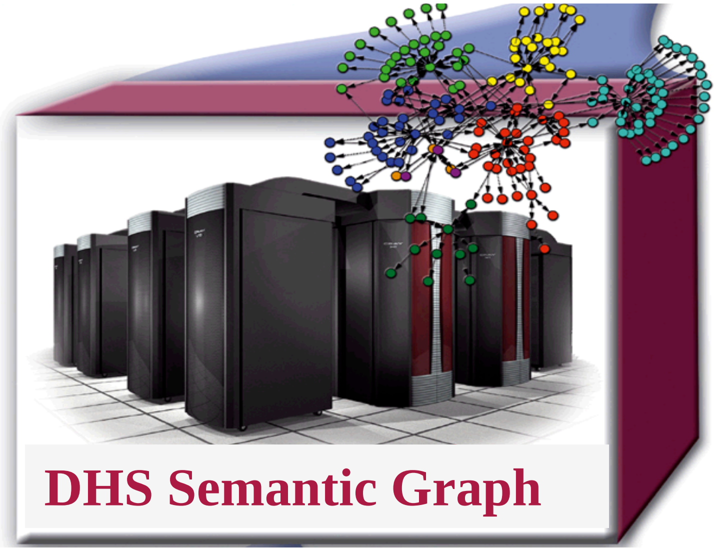
Predictor	ABNER	YAGI	KEX	LingPipe	NLProt	Voting + Meta-Classifier Score
Precision	25.9 %	33.3%	15.6 %	30.2 %	42.6 %	85.5%
Recall	58.2%	62.5 %	63.9%	73.1 %	49.9 %	67.3%

Intelligent Queries over Semantic Graphs

Processing of intelligent queries and advanced analysis of information in DHS presents a significant computational challenge.

Example Queries beyond Google

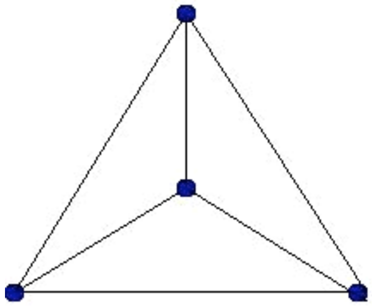
- Identify a minimum set of pathogens that are related to all the other pathogens (**Minimum Vertex Cover**);
- Discover a pattern of interest in the DB (**Sub-graph Isomorphism**);
- Find the largest group of cities so that every two cities are affected by a disease spreading from one city to another or enumerate all such groups (**Maximum or Maximal Clique**);
- Extract the maximum group of countries that have had the same disease spreading pattern this year as they had last year (**Maximum Common Subgraph**).



Example: Maximum Clique

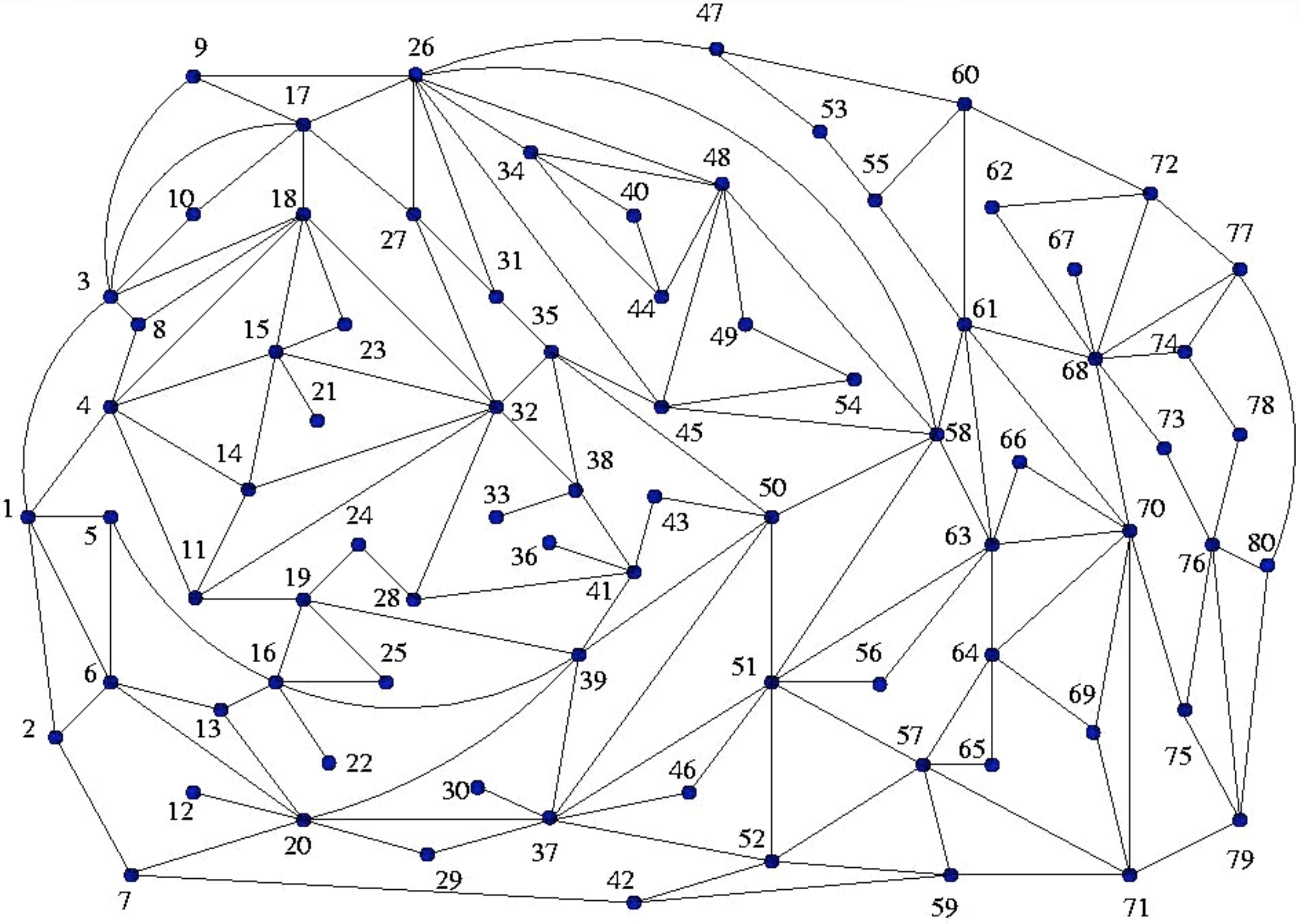
- A clique is a complete subgraph, for example, K_4:

- Finding maximum clique in a graph is *NP*-complete problem, and difficult even for small cliques on planar graphs

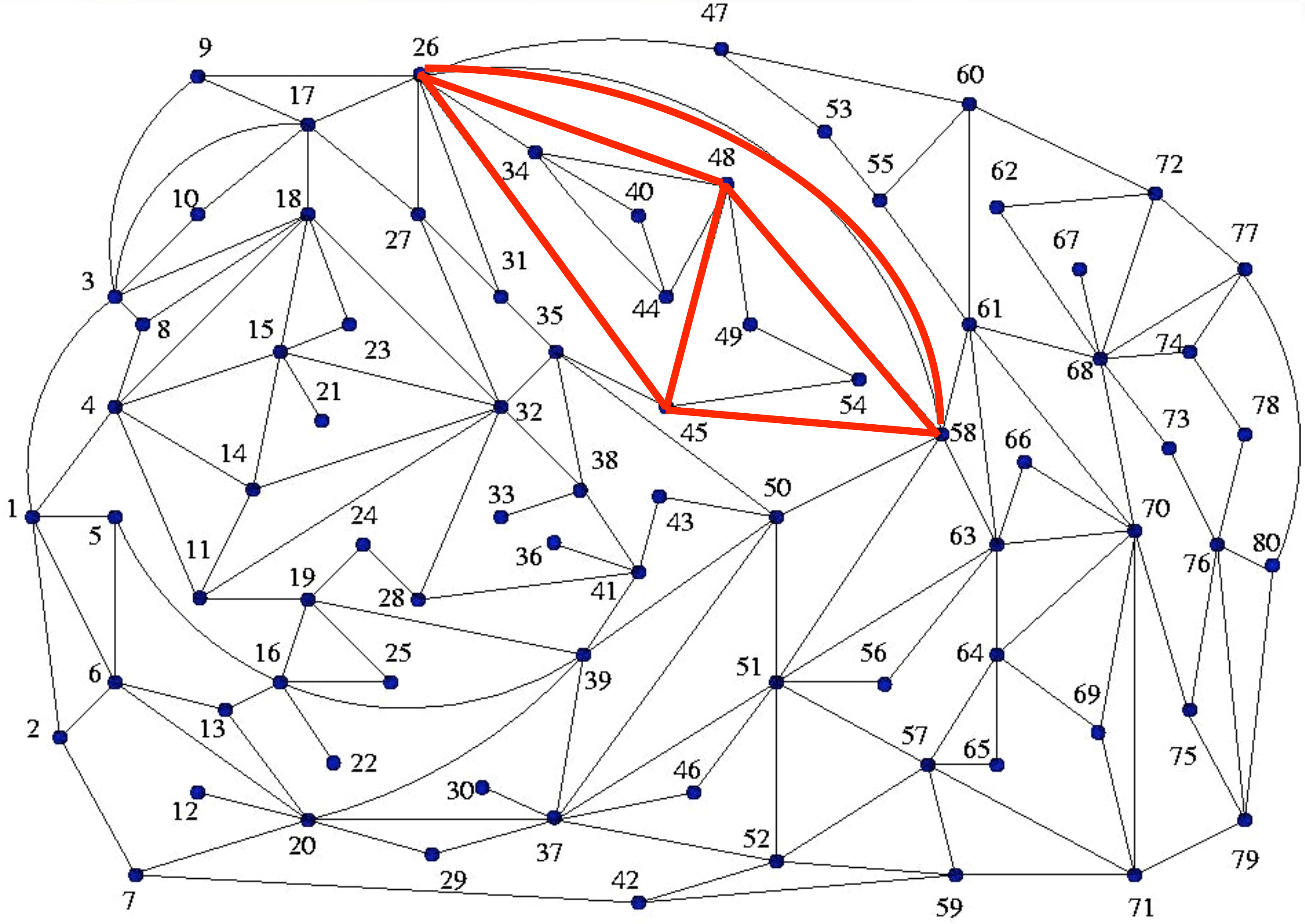


K_4

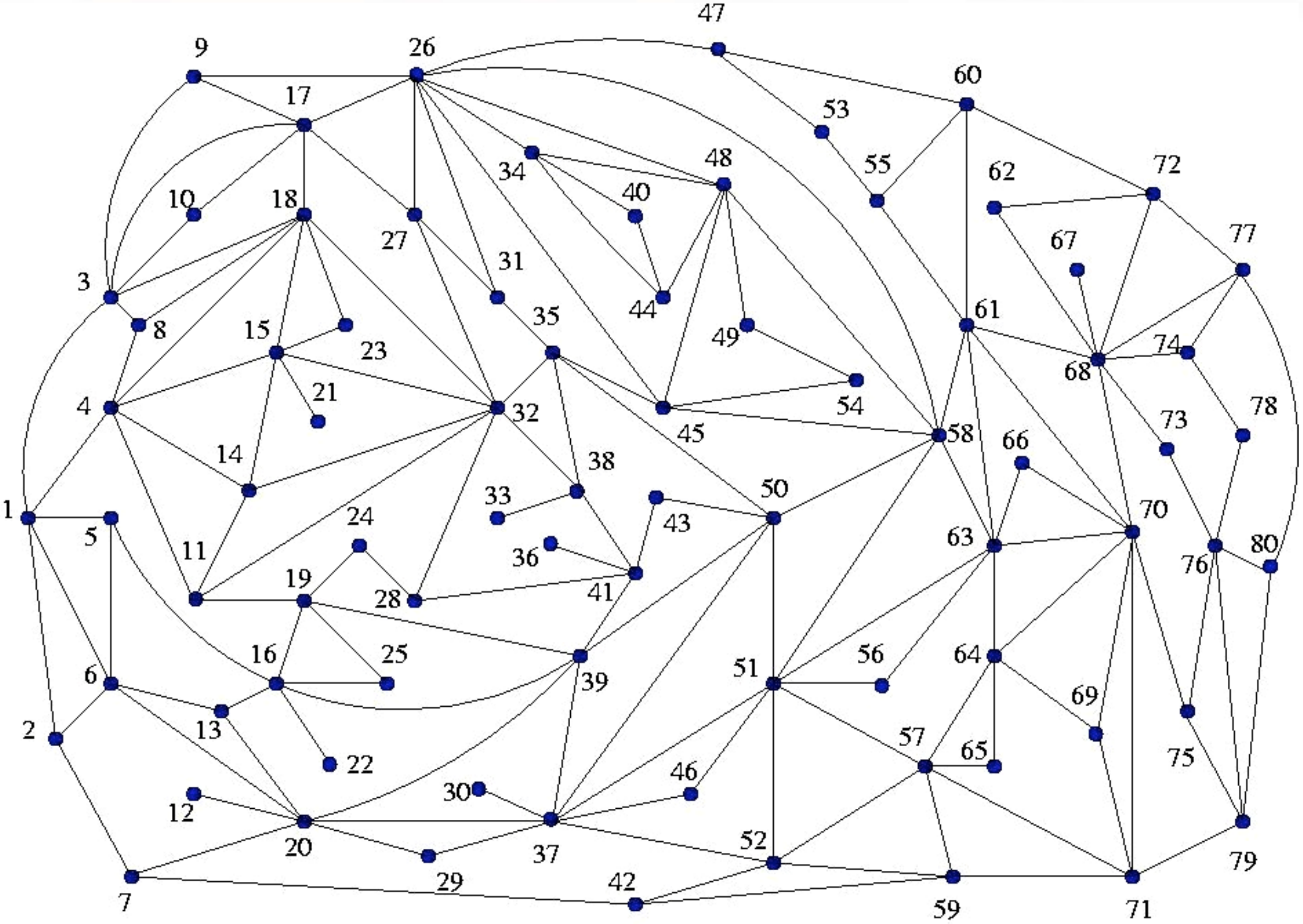
Does this graph contain K4?



Indeed it does!

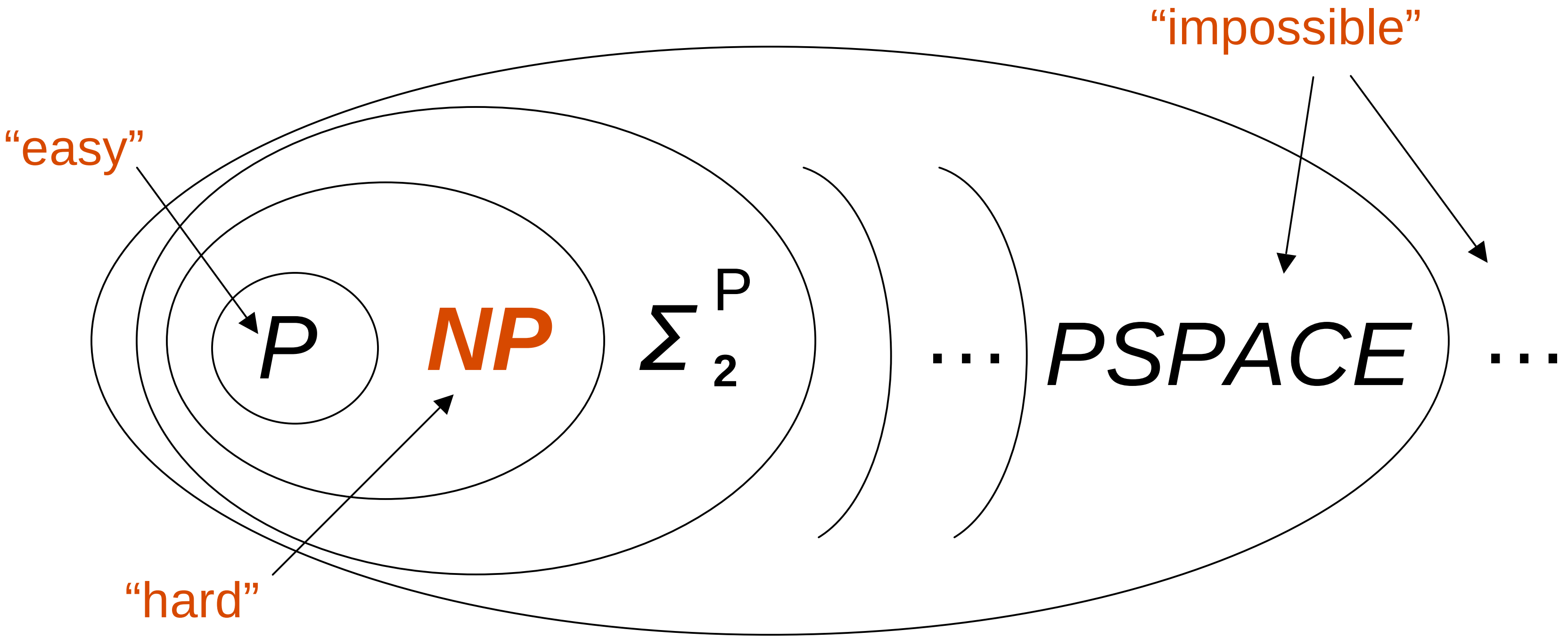


But, if it had not, what evidence would have been needed?



Classic Complexity Theory

- ***The Classic View:***

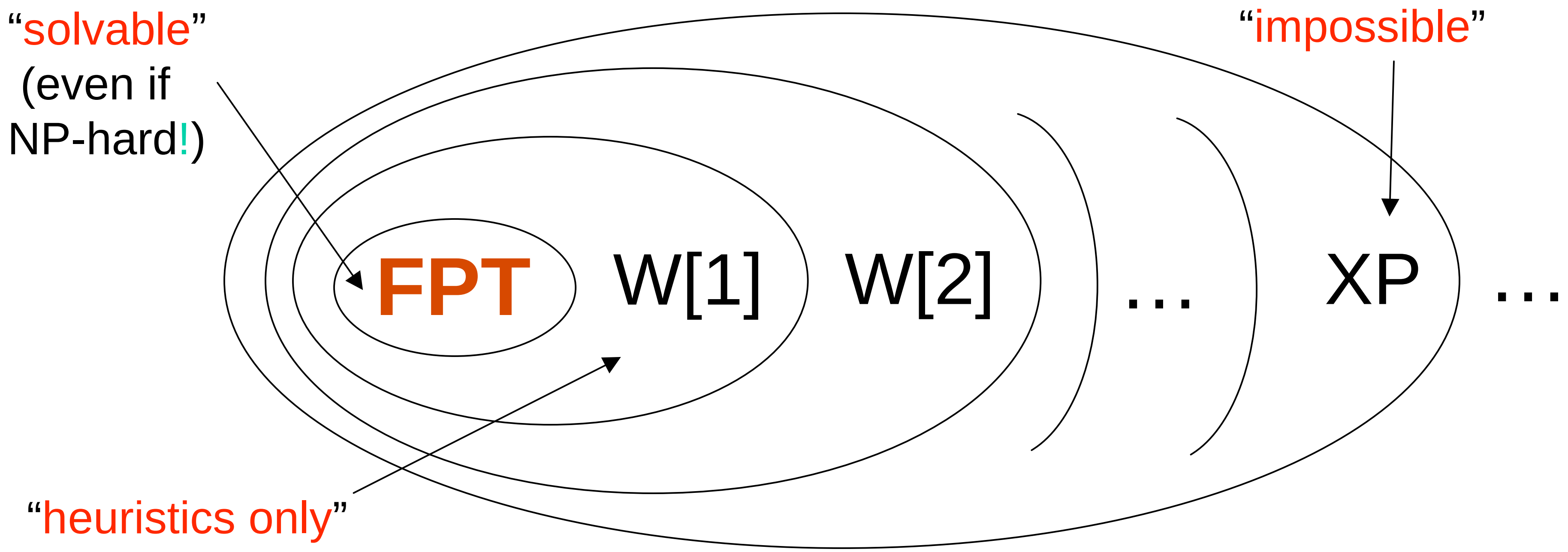


Parameter Sensitivity: Instance(n,k)

- Suppose our problem is, say, *NP*-complete.
- Consider an algorithm with a time bound such as $O(2^{k+n})$.
- And now one with a time bound more like $O(2^k + n)$.
- Both are exponential in parameter value(s).
- But what happens when *k* is fixed?

Parameterized Complexity Theory

Hence, the Parameterized View:



Fixed Parameter Tractability

- Fixed Parameter Tractability offers extremely efficient methods of reducing the search space for a certain subclass of *NP*-complete problems, known as FPT.

- FPT branching techniques also offer an effective method of parallelizing difficult problems:
  - Embarrassingly parallel
  - Little or no communication between processors

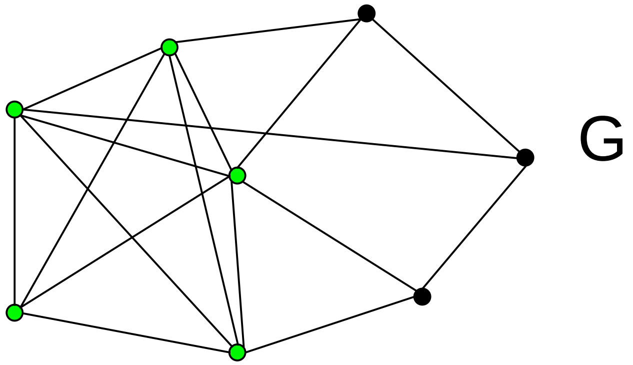
- These techniques have lead to the implementation of the world’s fastest codes for solving these well-known NP-complete problems.

Clique → Vertex Cover

Reduction:

- The Maximum Clique is not FPT
- Fortunately, Vertex Cover is FPT
- Vertex Cover is a complementary dual to Clique

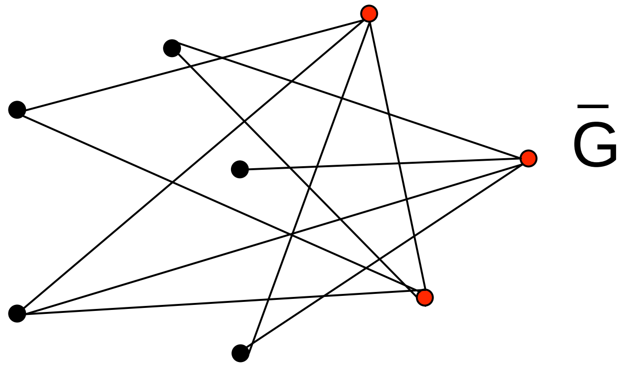
Maximum Clique (Size 5)



Vertex Cover - Major Steps:

- **preprocess** via degree structures
- **kernelize** to computational core
- parallel **branching** explores core
- **interleave** all three

Minimum Vertex Cover (Size 3)



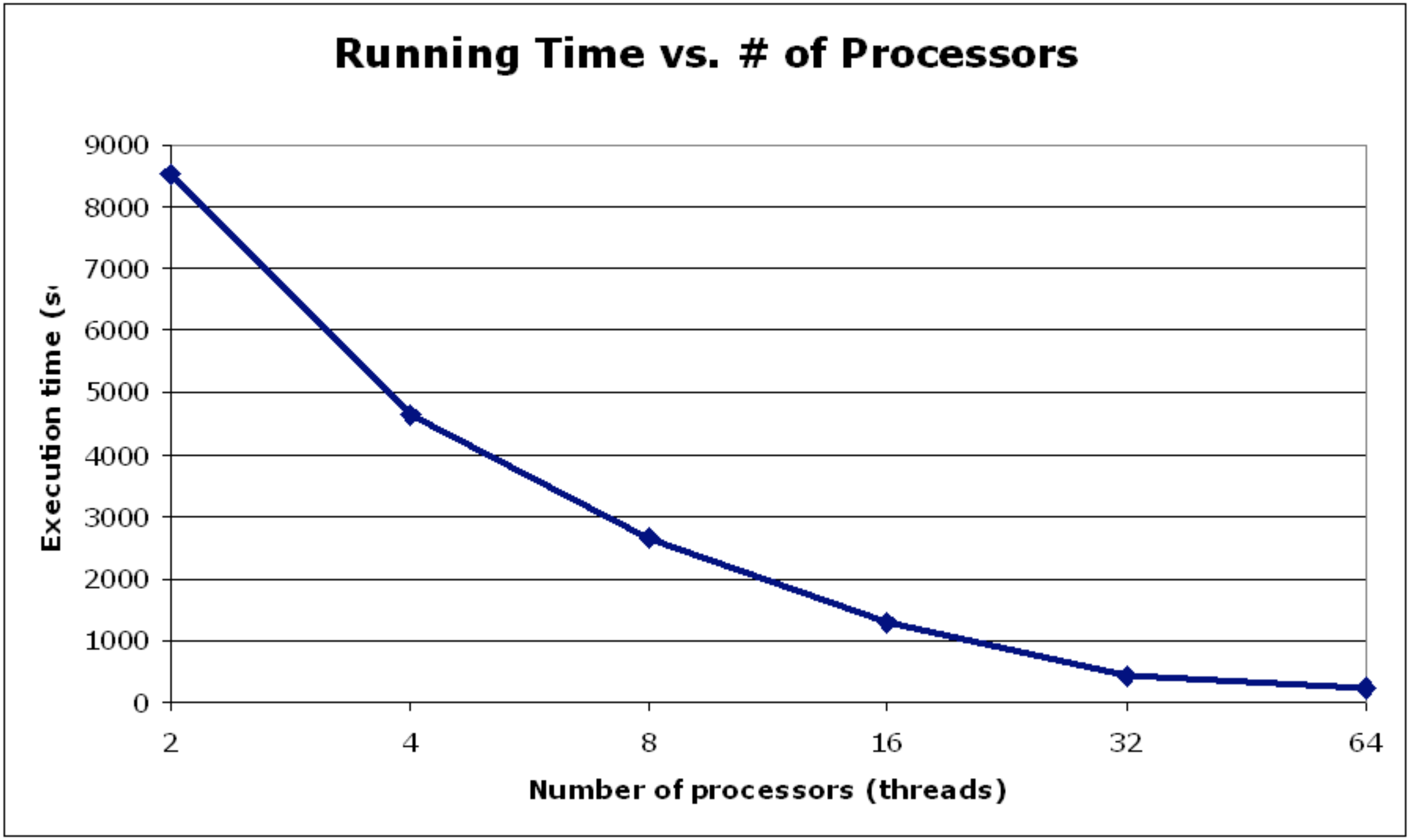
Performance Results

Graph Name	Graph Size	Cover Size	Instance Type	Sequential Kernelization	Sequential Branching	Parallel Branching	Dynamic Decomposition
Set-1	839	399	Yes	34 seconds	7 seconds	Not needed	Not needed
Set-2	839	398	No	34 seconds	141 minutes	82 minutes	20 minutes
Set-3	2466	2044	Yes	203 minutes	~ 5 days	~ 5 days	140 minutes
Set-4	2466	2043	No	203 minutes	6+ days	6+ days	620 minutes

So clique size is 422. **A direct assault ~ 2466^{422}.**

32 PEs @ 500MHz.
Load balancing is critical.
“No” is harder than “yes.”

Scalability



Graph Matching → Clique

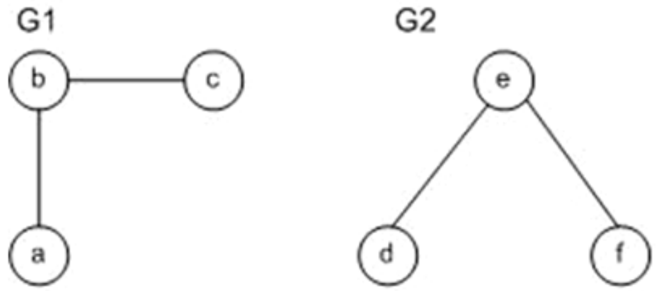
- Maximum Common Subgraph (MCS) and Subgraph Isomorphism are special cases of Graph Matching.

- Existing approaches to MCS:
  - Clique-based (Bron-Kerbosch, Robson); $O(1.19^{mn})$
  - Backtracking (McGregor, Krissinel); $O(m^{n+1}n)$
  - Dynamic programming (Akutsu) (trees of bounded degree)

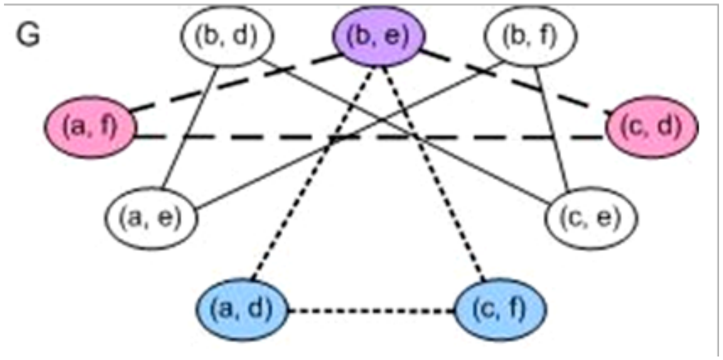
- MCS is not FPT. But we solve MCS by reducing it to Clique on the *association graph*.

- Our method is the fastest known on general graphs with $O((m+1)^n)$ but much better in practice since there are much less choices for branching than $(m+1)$

MCS of size 3



Clique of size 3



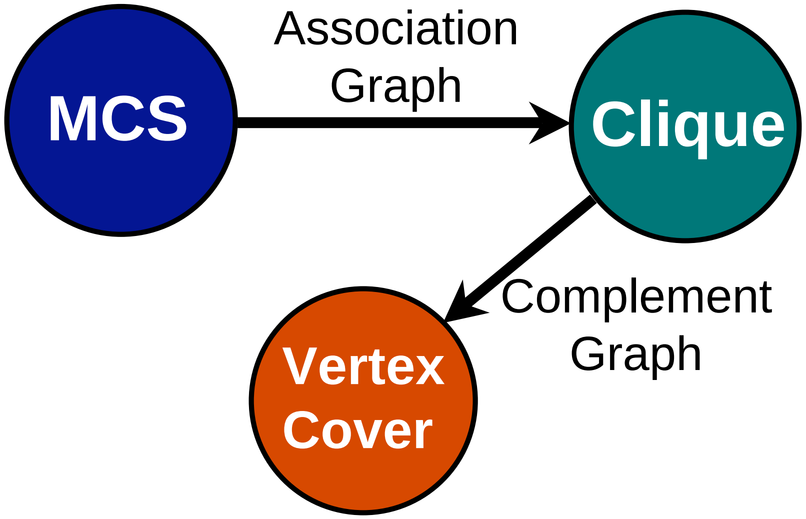
Association Graph

Scalable Algorithms for Semantic Graphs

Prototyped a library of scalable parallel graph matching algorithms for NP-hard graph problems with polynomial time solution.

Library Features:

- **Exact polynomial** solutions via **Fixed Parameter Tractability** (FPT) reduction:
  - Minimum Vertex Cover (VC)
  - Sub-graph Isomorphism (SI)
  - Maximum or Maximal Clique (Clique)
  - Maximum Common Subgraph (MCS)
- The **fastest and most scalable** (in problem size) than reported in literature.
- Supports different types of graphs: directed, undirected, labeled, and unlabeled.



Example Semantic Graph:
12,422 vertices and >100M edges
Maximum Clique: 399 vertices

Summary

***Goal*: Provide a capability for automated mapping of unstructured free text to Semantic Graph and for efficient query over Semantic Graph.**

- **Motivation**
  - The construction of the concept graphs from unstructured text is a very labor intensive and tedious task that requires automation.
  - Semantic graph queries are often NP-complete
- **Major accomplishments**
  - Intelligent text preprocessing
  - Advanced methods for concepts extraction, scoring, and mapping
  - Scalable graph algorithms over semantic graphs
- **Benefits**
  - Facilitate free text data feed to the BKC semantic graph.
  - Discover advanced knowledge from the semantic graph.